

**Claims**

What is claimed is:

- 1     1.     A method comprising:  
2         detecting an event associated with a device within a data processing system;  
3         initiating a platform-independent device removal sequence for said device in  
4     response to detecting said event associated with said device;  
5         virtually ejecting said device from said data processing system in response to  
6     initiating said platform-independent device removal sequence for said device; and  
7         servicing said event associated with said device in response to virtually  
8     ejecting said device from said data processing system.
- 1     2.     The method of claim 1, wherein initiating a platform-independent device  
2     removal sequence for said device in response to detecting said event associated with  
3     said device comprises generating a system control interrupt.
- 1     3.     The method of claim 2, wherein generating a system control interrupt  
2     comprises generating a system control interrupt utilizing an INT\_OUT command.
- 1     4.     The method of claim 1, wherein said device comprises a processor including a  
2     first processor core and a second processor core, said method further comprising  
3     operating said first processor core and said second processor core in a functional  
4     redundancy check mode.
- 1     5.     The method of claim 4, wherein detecting an event associated with a device  
2     within a data processing system comprises detecting an error within at least one of  
3     said first processor core and said second processor core.
- 1     6.     The method of claim 4, said method further comprising disabling interrupts to  
2     said processor in response to initiating a platform-independent device removal  
3     sequence for said device.

1 7. The method of claim 4, wherein operating said first processor core and said  
2 second processor core in a functional redundancy check mode comprises concurrently  
3 executing a thread on said first processor core and said second processor core, said  
4 method further comprising:

5 saving a context of said thread in response to initiating a platform-independent  
6 device removal sequence for said device.

1 8. The method of claim 5, said method further comprising disabling said  
2 functional redundancy check mode in response to detecting said error within at least  
3 one of said first processor core and said second processor core.

1 9. The method of claim 8, wherein servicing said event associated with said  
2 device in response to virtually ejecting said device from said data processing system  
3 comprises:

4 resetting said processor; and

5 re-enabling said functional redundancy check mode in response to resetting  
6 said processor.

1 10. The method of claim 1, said method further comprising:

2 initiating a platform-independent device insertion sequence for said device in  
3 response to servicing said event associated with said device; and

4 virtually inserting said device into said data processing system in response to  
5 initiating said platform-independent device insertion sequence for said device.

1 11. A machine-accessible medium having machine-executable instructions  
2 embodied therein which, when executed by a machine, causes said machine to  
3 perform a method comprising:

4 detecting an event associated with a device within a data processing system;

5 initiating a platform-independent device removal sequence for said device in  
6 response to detecting said event associated with said device;

7           virtually ejecting said device from said data processing system in response to  
8 initiating said platform-independent device removal sequence for said device; and  
9           servicing said event associated with said device in response to virtually  
10 ejecting said device from said data processing system.

1   12.    The machine-accessible medium of claim 11, wherein said device comprises a  
2 processor including a first processor core and a second processor core, said method  
3 further comprising operating said first processor core and said second processor core  
4 in a functional redundancy check mode.

1   13.    The machine-accessible medium of claim 12, wherein detecting an event  
2 associated with a device within a data processing system comprises detecting an error  
3 within at least one of said first processor core and said second processor core.

1   14.    The machine-accessible medium of claim 12, said method further comprising  
2 disabling interrupts to said processor in response to initiating a platform-independent  
3 device removal sequence for said device.

1   15.    The machine-accessible medium of claim 12, wherein operating said first  
2 processor core and said second processor core in a functional redundancy check mode  
3 comprises concurrently executing a thread on said first processor core and said second  
4 processor core, said method further comprising:

5           saving a context of said thread in response to initiating a platform-independent  
6 device removal sequence for said device.

1   16.    The machine-accessible medium of claim 13, said method further comprising  
2 disabling said functional redundancy check mode in response to detecting said error  
3 within said at least one of said first processor core and said second processor core.

1   17.    The machine-accessible medium of claim 16, wherein servicing said event  
2 associated with said device in response to virtually ejecting said device from said data  
3 processing system comprises:

4           resetting said processor; and  
5           re-enabling said functional redundancy check mode in response to resetting  
6           said processor.

1    18.    A data processing system comprising:  
2           a processor; and  
3           a memory coupled to said processor, said memory including a firmware  
4           interface to initiate a platform-independent device removal sequence for said  
5           processor in response to a detection of an event associated with said processor, to  
6           virtually eject said processor from said data processing system in response to an  
7           initiation of a platform-independent device removal sequence for said processor, and  
8           to service said event associated with said processor in response to a virtual ejection of  
9           said processor from said data processing system.

1    19.    The data processing system of claim 18, wherein said firmware interface to  
2           initiate a platform-independent device removal sequence for said processor in  
3           response to a detection of an event associated with said processor comprises a  
4           firmware interface to generate a system control interrupt.

1    20.    The data processing system of claim 19, wherein said firmware interface to  
2           generate a system control interrupt comprise a firmware interface to generate a system  
3           control interrupt utilizing an INT\_OUT command.

1    21.    The data processing system of claim 18, wherein said processor comprises a  
2           first processor core and a second processor core, said data processing system further  
3           comprising functional redundancy check logic to operate said first processor core and  
4           said second processor core in a functional redundancy check mode.

1    22.    The data processing system of claim 21, wherein said firmware interface to  
2           initiate a platform-independent device removal sequence for said processor in  
3           response to a detection of an event associated with said processor comprises a  
4           firmware interface to initiate a platform-independent device removal sequence for

5 said processor in response to a detection of an error within at least one of said first  
6 processor core and said second processor core.

1 23. The data processing system of claim 21, said memory further including an  
2 operating system to disable interrupts to said processor in response to an initiation of a  
3 platform-independent device removal sequence for said processor.

1 24. The data processing system of claim 21, wherein  
2 said functional redundancy check logic to operate said first processor core and  
3 said second processor core in a functional redundancy check mode comprises  
4 functional redundancy check logic to concurrently execute a thread on said first  
5 processor core and said second processor core; and  
6 said memory further includes an operating system to save a context of said  
7 thread in response to an initiation of a platform-independent device removal sequence  
8 for said processor.

1 25. The data processing system of claim 21, wherein said firmware interface  
2 further comprises a firmware interface to disable said functional redundancy check  
3 mode in response to said detection of an event associated with said processor.

1 26. The data processing system of claim 25, wherein said firmware interface to  
2 service said event associated with said processor in response to a virtual ejection of  
3 said processor from said data processing system comprises a firmware interface to  
4 reset said processor and to re-enable said functional redundancy check mode in  
5 response to a reset of said processor.

1 27. The data processing system of claim 18, wherein said firmware interface  
2 further comprises a firmware interface to initiate a platform-independent device  
3 insertion sequence for said processor in response to a servicing of said event  
4 associated with said processor and to virtually insert said processor into said data  
5 processing system in response to an initiation of a platform-independent device  
6 insertion sequence for said processor.

1 28. An apparatus comprising:

2 a first firmware interface to detect an event associated with a processor within  
3 a data processing system;

4 a second firmware interface to initiate a platform-independent device removal  
5 sequence for said processor in response to a detection of said event associated with  
6 said processor;

7 a third firmware interface to virtually eject said processor from said data  
8 processing system in response to an initiation of a platform-independent device  
9 removal sequence for said processor; and

10 a fourth firmware interface to service said event associated with said processor  
11 in response to a virtual ejection of said processor from said data processing system.

1 29. The apparatus of claim 28, wherein said second firmware interface to initiate a  
2 platform-independent device removal sequence for said processor in response to a  
3 detection of said event associated with said processor comprises a firmware interface  
4 to generate a system control interrupt.

1 30. The apparatus of claim 28, said apparatus further comprising:

2 a fifth firmware interface to initiate a platform-independent device insertion  
3 sequence for said processor in response to a servicing of said event associated with  
4 said processor; and

5 a sixth firmware interface to virtually insert said processor into said data  
6 processing system in response to an initiation of a platform-independent device  
7 insertion sequence for said processor.